

1 CLAIMS

2 1. A method of parsing an Extensible Markup Language (XML) data
3 stream comprising:

4 defining a plurality of states, individual states being associated with
5 individual elements of an XML data stream;

6 associating one or more rules with each state;

7 receiving an XML data stream;

8 evaluating the XML data stream against one or more of the rules for
9 individual elements contained in the XML data stream; and

10 disregarding associated portions of the XML data stream if any of the rules
11 that are associated with those portions are violated.

12
13 2. The method of claim 1, wherein the one or more rules relate to a
14 schema of the XML data stream.

15
16 3. The method of claim 1, wherein:
17 the defining of the plurality of states comprises defining one or more
18 schema modules that are configured to track one or more states of the XML data
19 stream; and

20 the evaluating comprises using the one or more schema modules to evaluate
21 the XML data stream against one or more schema-based rules.

1 4. The method of claim 1, wherein the defining of the plurality of states
2 comprises defining one or more schema modules that are configured to track one
3 or more states of the XML data stream, each schema module being associated with
4 at least one request type that defines the XML data stream.

5
6 5. The method of claim 4, wherein the request type is a WebDAV
7 request type.

8
9 6. The method of claim 5, wherein the WebDAV request type is a
10 PROPFIND request.

11
12 7. The method of claim 5, wherein the WebDAV request type is a
13 PROPPATCH request.

14
15 8. The method of claim 5, wherein the WebDAV request type is a
16 SEARCH request.

17
18 9. The method of claim 5, wherein the WebDAV request type is one of
19 a LOCK and UNLOCK request.

20
21 10. The method of claim 1 further comprising defining one or more
22 rules that relate to an element's contents.

1 11. The method of claim 10, wherein said one or more rules that relate
2 to an element's contents define which elements can be contained within other
3 elements.

4
5 12. The method of claim 11, wherein if a rule that defines which
6 elements can be contained within other elements is violated, disregarding
7 associated portions of the XML data stream until a close tag is received for an
8 element that violates the rule.

9
10 13. A computer-readable medium having a program thereon which,
11 when executed by a computer, performs the steps of claim 1.

12
13 14. A method of parsing an Extensible Markup Language (XML) data
14 stream comprising:

15 defining a schema module that is associated with an HTTP request type that
16 is received from a client, the schema module having a function that determines
17 whether an XML data stream conforms to a given schema that is associated with
18 the HTTP request type;

19 evaluating an XML data stream with the schema module; and

20 disregarding a portion of the XML data stream if it does not conform to the
21 given schema.

1 15. The method of claim 14, wherein said defining of the schema
2 module comprises defining a plurality of schema modules, individual schema
3 modules being associated with different HTTP request types.

4
5 16. The method of claim 14, wherein said function determines whether
6 there are any unauthorized elements that appear in a client's request.

7
8 17. The method of claim 14, wherein said function determines whether
9 there are any unauthorized elements that appear in a client's request; said
10 disregarding comprising disregarding said XML data stream portion until a close
11 tag is received for an unauthorized element.

12
13 18. The method of claim 14, wherein said HTTP request type comprises
14 a WebDAV request type.

15
16 19. The method of claim 18, wherein said WebDAV request type
17 comprises a PROPFIND request.

18
19 20. The method of claim 18, wherein said WebDAV request type
20 comprises a PROPPATCH request.

21
22 21. The method of claim 18, wherein said WebDAV request type
23 comprises a SEARCH request.

1 22. The method of claim 18, wherein said WebDAV request type
2 comprises one of a LOCK and UNLOCK request.

3
4 23. A computer-readable medium having a program thereon which,
5 when executed by a computer, performs the steps of claim 14.

6
7 ~~24.~~ An Extensible Markup Language (XML) parsing system
8 comprising:

9 a parser configured to receive an XML data stream and generate a series of
10 calls as it parses the XML data stream;

11 a node factory communicatively associated with the parser and configured
12 to receive the parser's calls and responsive thereto construct a representation of the
13 XML data stream that the parser is parsing; and

14 a schema module communicatively associated with the node factory and
15 configured to evaluate the node factory's representation of the XML data stream
16 and determine whether it conforms to a known schema.

17
18 25. The parsing system of claim 24, wherein said parsing system
19 comprises a plurality of schema modules, each schema module being associated
20 with a different known schema.

21
22 26. The parsing system of claim 24, wherein the schema module
23 corresponds to an HTTP request type.
24
25

1 27. The parsing system of claim 24, wherein said parsing system
2 comprises a plurality of schema modules, each schema module being associated
3 with a different known schema and corresponding to a different HTTP request
4 type.

5
6 28. The parsing system of claim 27, wherein at least one of the different
7 HTTP request types is a WebDAV request.

8
9 29. The parsing system of claim 24, wherein the schema module is
10 configured to ignore an XML data stream portion that does not conform to the
11 known schema.

12
13 30. An Extensible Markup Language (XML) parsing system
14 comprising:

15 a collection of schema modules, each of which being configured to evaluate
16 a different schema that is associated with an XML data stream; and

17 a plurality of states associated with each schema module, individual states
18 of a schema module defining a schema requirement relating to a particular element
19 that is evaluated by that schema module.

20
21 31. The parsing system of claim 30, wherein each schema module is
22 associated with a different HTTP request and is configured to evaluate a schema
23 that is associated with the HTTP request with which is it associated.

1 32. The parsing system of claim 31, wherein at least one of the HTTP
2 requests is a WebDAV request.

3
4 33. The parsing system of claim 31, wherein each of the HTTP requests
5 is a WebDAV request.

6 Add
7 C17

09361372-0726991320